

A CRM APPLICATION FOR LAPTOP RENTALS

By

Shaik Meharaj

21501A05F9@pvpsit.ac.in

Project Abstract

In today's fast-paced digital era, the demand for flexible, short-term access to high-quality technology has surged, particularly within the laptop rental market. Companies, students, and professionals alike are increasingly turning to rental services to meet their temporary technology needs without the substantial financial commitment of purchasing new equipment. Recognizing this growing demand, the CRM Application for Laptop Rentals has been developed to deliver a seamless, customer-centric solution that optimizes the entire rental process from start to finish.

The primary objective of this CRM application is to leverage advanced customer relationship management (CRM) practices to enhance customer experiences, streamline store operations, and improve overall efficiency within the laptop rental industry. By integrating a robust CRM system into the rental process, businesses can ensure that every customer interaction is personalized, efficient, and geared toward long-term customer satisfaction and loyalty.

Key Features and Functionalities:

- Enhanced Customer Experience:** The CRM application is designed to provide a highly personalized and responsive customer experience. It allows businesses to maintain detailed profiles of each customer, tracking preferences, rental history, and feedback. This information enables businesses to anticipate customer needs, offer tailored recommendations, and ensure that each rental experience meets or exceeds expectations.
- Optimized Store Operations:** The application streamlines various operational aspects of the rental business, from inventory management to order processing. Real-time inventory tracking ensures that the availability of laptops is always up to date, minimizing the risk of overbooking or stockouts. Additionally, the application automates key processes such as invoicing, payment processing, and order confirmation, reducing the workload on staff and allowing for quicker turnaround times.
- Improved Efficiency and Productivity:** By automating routine tasks and providing a centralized platform for managing all customer interactions, the CRM application significantly improves overall efficiency. Staff can quickly access relevant customer information, track ongoing orders, and address any issues that may arise during the rental period. The system also generates detailed reports and analytics, providing valuable insights into business performance and customer behaviour, which can be used to make informed decisions and drive growth.
- Effective Customer Relationship Management:** Beyond managing rentals, the CRM application plays a crucial role in building and maintaining strong customer

relationships. It includes integrated communication tools that allow businesses to reach out to potential and existing customers through targeted email campaigns. These campaigns can be used to promote special offers, share useful content, and keep customers informed about new products and services, thereby fostering a sense of loyalty and encouraging repeat business.

5. **Scalability and Customization:** The CRM application is built with scalability in mind, making it suitable for businesses of all sizes, from small startups to large enterprises. It offers a high degree of customization, allowing businesses to tailor the system to their specific needs and preferences. Whether it's adding new features, integrating third-party tools, or adjusting workflows, the application can be adapted to meet the evolving demands of the business.

Conclusion:

The CRM Application for Laptop Rentals represents a significant advancement in the way rental services are managed and delivered. By combining the power of CRM with innovative technology solutions, this application not only enhances the customer experience but also drives operational efficiency and business growth. As the laptop rental market continues to expand, this CRM application provides businesses with the tools they need to stay competitive, build lasting customer relationships, and achieve long-term success.

INDEX Page

Topics	Page no
1. Creating a Developer Account in Salesforce	5-6
2. Salesforce Objects and Custom Object Creation	7-9
3. Understanding and Creating Tabs in Salesforce	10-12
4. Creating a Lightning App in Salesforce	13-14
5. Fields in Salesforce	15-37
6. Creating a Validation Rule for Phone Number and Email in the Consumer Object	38
7. Profiles in Salesforce	39-40
8. Roles and Hierarchy in Salesforce	41
9. Creating Users in Salesforce	42-43
10. Creating a Flow in Salesforce	44-54
11. Creating a Trigger in Salesforce	55-58
12. Salesforce Reports	59-63
13. Salesforce Dashboards	64-66

Topic 1: Creating a Developer Account in Salesforce

Creating a Salesforce Developer account is the first and crucial step to accessing Salesforce's robust development tools and environment. This account allows you to explore, build, and customize applications on the Salesforce platform. Here's a step-by-step guide to creating and activating your Salesforce Developer account:

1. Signing Up for a Developer Account

To begin, you need to create a Salesforce Developer account by following these steps:

1. Visit the Salesforce Developer Sign-Up Page:

- Navigate to <https://developer.salesforce.com/signup> using your web browser.

2. Fill Out the Sign-Up Form:

- **First Name & Last Name:** Enter your full name. This information will be used for your account.
- **Email:** Provide a valid email address. This will be used for communication and account verification purposes.
- **Role:** Select "Developer" from the dropdown menu. This ensures that the account is tailored to your development needs.
- **Company:** Enter the name of your college. This is essential as it will be part of your username.
- **Country:** Choose "India" from the list of countries.
- **Postal Code:** Enter your local postal code.
- **Username:** Create a unique username. This should be a combination of your name and the company (college) name. The format typically looks like `username@organization.com`. Note that this username does not have to be an actual email address, so you can create any username in the mentioned format.

3. Complete the Sign-Up Process:

- After filling in all the required details, click on the "Sign me up" button. This submits your information and triggers the account creation process.

2. Account Activation

Once you've completed the sign-up form, the next step is to activate your Salesforce Developer account. Here's how:

1. Check Your Email for Account Verification:

- Go to the inbox of the email address you provided during sign-up. Salesforce will send you a verification email within 5-10 minutes. The subject line will typically include a message asking you to verify your account.

2. Verify Your Account:

- Open the email from Salesforce and click on the "Verify Account" link provided. This link will take you to the account activation page.

3. Set Up Your Password:

- On the account verification page, you will be prompted to create a password. Choose a strong password that meets Salesforce's security requirements.
- You will also need to select and answer a security question. This adds an extra layer of protection to your account, allowing you to recover access in case you forget your password.

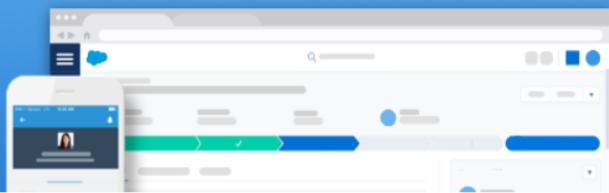
4. Access the Salesforce Setup Page:

- After setting your password and answering the security question, click on "Change Password." You will be automatically redirected to your Salesforce setup page.
- The setup page is the central hub where you can manage your Salesforce environment. From here, you can start exploring the platform, customize your org, create objects, fields, and begin developing applications.

Conclusion

Creating and activating a Salesforce Developer account is a straightforward process that gives you access to a powerful suite of tools and features. This account is essential for anyone looking to explore Salesforce's capabilities, whether for learning, development, or building real-world applications. Once your account is set up, you can begin customizing your Salesforce environment, working on projects, and leveraging the platform's extensive resources to enhance your skills as a developer.

Thanks for signing up with Salesforce!



Click below to verify your account.

Verify Account

To easily log in later, save this URL:

<https://prasadvpotlurisiddharth-1af-dev-ed.develop.my.salesforce.com>

Username:
shaikmeharaj@123gmail.com

Again, welcome to Salesforce!

Topic 2: Salesforce Objects and Custom Object Creation

Salesforce objects serve as the fundamental building blocks of the Salesforce platform, functioning as database tables that store data specific to an organization. Understanding these objects is crucial for managing and manipulating data effectively within Salesforce. In this section, we'll explore the types of Salesforce objects and provide a detailed guide on creating custom objects that are tailored to the specific needs of a laptop rental CRM application.

Types of Salesforce Objects

Salesforce objects are classified into two primary types:

1. Standard Objects:

- **Definition:** Standard objects are pre-defined objects provided by Salesforce that are included in every Salesforce organization. These objects come with built-in functionality and are essential for many core features of the Salesforce platform.
- **Examples:**
 - **Accounts:** Represents an individual account, which could be a customer, partner, or competitor.
 - **Contacts:** Holds information about individuals associated with an account.
 - **Opportunities:** Tracks sales opportunities, including the stages of a deal.
 - **Leads:** Manages potential customers who are not yet qualified.
 - **Users:** Stores data about individual users who have access to Salesforce.
 - **Contracts, Reports, Dashboards:** Used to manage agreements, generate data insights, and visualize performance metrics.

2. Custom Objects:

- **Definition:** Custom objects are user-defined objects created to store data unique and essential to a particular organization. They form the heart of any application on Salesforce, enabling organizations to manage and share data that doesn't fit into the standard objects.
- **Purpose:** Custom objects are essential for tailoring Salesforce to specific business needs. For instance, in a laptop rental business, custom objects could be used to track inventory, manage customer information, or record rental transactions.

Creating Custom Objects for a Laptop Rental CRM

To effectively manage a laptop rental business on Salesforce, several custom objects need to be created. These objects will help in tracking laptops, managing customer information, handling bookings, and overseeing the billing process. Below is a step-by-step guide on how to create these custom objects:

1. Creating the “Total Laptops” Object:

- **Purpose:** This object will be used to store information about the total inventory of laptops available for rent.
- **Steps:**
 1. Navigate to the **Setup** page.
 2. Click on **Object Manager**.
 3. Click on **Create** and then select **Custom Object**.
 4. Enter the **Label Name** as "Total Laptops".
 5. The **Plural Label Name** will also be "Total Laptops".
 6. Set the **Record Name** as "Total Laptops" and choose **Data Type** as "Text".
 7. Enable options for **Allow Reports**, **Allow Search**, and **Track Field History**.
 8. Click **Save**.

The screenshot shows the Salesforce Setup > Object Manager interface. A new custom object named 'Total Laptops' is being created. The 'Details' tab is selected, showing the following configuration:

Field	Value	Options
Description		
API Name	Total_Laptops__c	Enable Reports <input checked="" type="checkbox"/>
Custom	✓	Track Activities <input checked="" type="checkbox"/>
Singular Label	Total Laptops	Track Field History <input checked="" type="checkbox"/>
Plural Label	Total Laptops	Deployment Status Deployed
		Help Settings Standard salesforce.com Help Window

The left sidebar lists other configuration tabs: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts.

2. Creating the “Consumer” Object:

- **Purpose:** This object will store information about the consumers who rent laptops.
- **Steps:**
 1. From the **Setup** page, go to **Object Manager**.
 2. Click on **Create** and choose **Custom Object**.
 3. Enter the **Label Name** as "Consumer".
 4. The **Plural Label Name** will also be "Consumer".
 5. Set the **Record Name** as "Consumer Name" and choose **Data Type** as "Name".
 6. Enable **Allow Reports, Allow Search, and Track Field History**.
 7. Click **Save**.

The screenshot shows the 'consumer' custom object setup page in the Salesforce Object Manager. The left sidebar lists various configuration tabs: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, and List View Button Layout. The main content area is titled 'Details' and contains fields for Description, API Name (consumer_c), Singular Label (consumer), and Plural Label (consumer). On the right side, there are checkboxes for Enable Reports (checked), Track Activities, Track Field History (checked), Deployment Status (Deployed), Help Settings, and a link to Standard salesforce.com Help Window.

Details	
Description	
API Name	consumer_c
Singular Label	consumer
Plural Label	consumer
Enable Reports	<input checked="" type="checkbox"/>
Track Activities	
Track Field History	<input checked="" type="checkbox"/>
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

3. Creating the “Laptop Bookings” Object:

- **Purpose:** This object will track each booking made by customers, detailing the laptops rented, the rental period, and related information.
- **Steps:**
 1. Go to **Object Manager** from the **Setup** page.
 2. Click on **Create** and then select **Custom Object**.
 3. Enter the **Label Name** as "Laptop Bookings".
 4. The **Plural Label Name** will be "Laptop Bookings".

5. Set the **Record Name** as "Laptop Bookings" and choose **Data Type** as "Name".
6. Enable **Allow Reports, Allow Search, and Track Field History**.
7. Click **Save**.

Laptop Bookings

Details

Description

API Name: Laptop_Bookings_c

Custom: ✓

Singular Label: Laptop Bookings

Plural Label: Laptop Bookings

Enable Reports: ✓

Track Activities

Track Field History: ✓

Deployment Status: Deployed

Help Settings

Standard salesforce.com Help Window

4. Creating the “Billing Process” Object:

- **Purpose:** This object will manage the billing and payment processing for laptop rentals.
- **Steps:**
 1. From the **Setup** page, navigate to **Object Manager**.
 2. Click on **Create** and select **Custom Object**.
 3. Enter the **Label Name** as "Billing Process".
 4. The **Plural Label Name** will also be "Billing Process".
 5. Set the **Record Name** as "Billing Process Name" and choose **Data Type** as "Name".
 6. Enable **Allow Reports, Allow Search, and Track Field History**.
 7. Click **Save**.

The screenshot shows the 'Object Manager' interface for the 'Billing Process' object. On the left, a sidebar lists various configuration options: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, and List View Button Layout. The main panel is titled 'Details' and contains sections for Description, API Name (set to 'Billing_Process__c'), Custom status (checked), Singular Label ('Billing Process'), and Plural Label ('Billing Process'). To the right, there are sections for Enable Reports (checked), Track Activities, Track Field History (checked), Deployment Status ('Deployed'), Help Settings, and a link to 'Standard salesforce.com Help Window'.

Conclusion

Custom objects in Salesforce provide a flexible and powerful way to store and manage data that is specific to an organization's needs. By creating custom objects such as "Total Laptops," "Consumer," "Laptop Bookings," and "Billing Process," a laptop rental business can effectively manage its inventory, customer data, rental transactions, and billing processes. These objects form the backbone of the CRM application, enabling the organization to streamline operations, improve customer relationships, and drive business success. As you continue to build out your Salesforce environment.

Topic 3: Understanding and Creating Tabs in Salesforce

Tabs in Salesforce provide a user interface that enables users to view and manage records of different objects within the platform. They serve as a crucial navigation component, allowing users to access and interact with various parts of Salesforce efficiently. In this section, we'll explore the types of tabs available in Salesforce and provide detailed instructions for creating custom tabs.

Types of Tabs

1. Custom Tabs:

- **Definition:** Custom object tabs are designed to provide a user interface for custom objects created within Salesforce. They allow users to access and manage records of custom objects, similar to how they interact with standard Salesforce objects such as Accounts and Contacts.
- **Appearance:** These tabs look and behave like standard Salesforce tabs and can be tailored to fit the needs of the specific custom application being developed.

2. Web Tabs:

- **Definition:** Web Tabs display web content or applications directly within the Salesforce interface. They are useful for integrating external web-based applications or content into the Salesforce environment, providing a seamless user experience without needing to leave the Salesforce platform.
- **Usage:** Commonly used to provide quick access to frequently used web resources or applications.

3. Visualforce Tabs:

- **Definition:** Visualforce Tabs are used to display Visualforce pages, which are custom user interfaces created using Visualforce, Salesforce's markup language. These tabs allow users to interact with custom pages and applications built using Visualforce.
- **Appearance:** Visualforce tabs appear like standard Salesforce tabs and integrate custom Visualforce pages into the Salesforce interface.

4. Lightning Component Tabs:

- **Definition:** Lightning Component Tabs enable users to add Lightning components to the navigation menu in Lightning Experience and the Salesforce mobile app. These tabs enhance the user experience by providing access to interactive Lightning components directly from the navigation bar.

- **Usage:** Useful for incorporating custom Lightning components into the navigation for easy access.

5. Lightning Page Tabs:

- **Definition:** Lightning Page Tabs allow users to add Lightning Pages to the mobile app navigation menu. Unlike other custom tabs, Lightning Page Tabs do not appear on the All-Tabs page or in the Available Tabs list when customizing tabs for apps.
- **Usage:** Ideal for organizing and accessing custom Lightning Pages within the mobile app navigation.

Creating a Custom Tab

To create a custom tab for your custom object, follow these steps:

1. Navigate to the Setup Page:

- Go to the Salesforce Setup page.

2. Access the Tabs Configuration:

- Type “Tabs” in the Quick Find bar and select **Tabs** from the dropdown menu.

3. Create a New Tab:

- Click on **New** under the Custom Object Tabs section.

4. Select Object:

- Choose the custom object for which you want to create a tab. For this example, select **Total Laptops**.

5. Select Tab Style:

- Choose a tab style that visually represents your custom object. This style will be displayed in the Salesforce navigation bar.

6. Configure Tab Settings:

- Click **Next** to proceed to the settings page.
- Under **Add to Profiles**, keep the default setting to ensure the tab is available to all profiles. You can customize this setting based on your requirements.
- For **Add to Custom App**, uncheck the option to include the tab in the custom app if you do not want it added.

- Ensure that the checkbox **Append tab to users' existing personal customizations** is selected to allow users to add this tab to their personal navigation bar.

7. Save the Tab:

- Click **Save** to create and finalize the custom tab.

Activity 2: Creating Remaining Tabs

Follow the same steps outlined above to create tabs for the remaining custom objects in your CRM application:

1. Create Tab for “Consumer” Object:

- Repeat the process, selecting **Consumer** as the custom object.
- Choose the appropriate tab style and configure the tab settings as needed.

2. Create Tab for “Laptop Bookings” Object:

- Repeat the process, selecting **Laptop Bookings** as the custom object.
- Choose the appropriate tab style and configure the tab settings as needed.

3. Create Tab for “Billing Process” Object:

- Repeat the process, selecting **Billing Process** as the custom object.
- Choose the appropriate tab style and configure the tab settings as needed.

By creating these tabs, you ensure that users have easy access to the custom objects and can efficiently manage and interact with the data specific to your laptop rental CRM application. Each tab will provide a clear and intuitive interface for accessing records, enhancing the overall usability of the Salesforce platform.

The screenshot shows the Salesforce Setup page for creating a new custom tab. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar says 'Search Setup'. On the left, a sidebar has 'User Interface' expanded, with 'Rename Tabs and Labels' and 'Tabs' selected. A message at the bottom says 'Didn't find what you're looking for? Try using Global Search.' The main content area is titled 'Custom Tabs' and contains a sub-section 'Custom Object Tabs'. It shows a table with four rows:

Action	Label	Tab Style
Edit Del	Billing_Process	Bank
Edit Del	consumer	People
Edit Del	Laptop Bookings	Shopping Cart
Edit Del	Total Laptops	Laptop

Below this is another section titled 'Web Tabs'.

Topic 4: Creating a Lightning App in Salesforce

Lightning apps in Salesforce provide a streamlined way to bundle related objects, tabs, and other components into a cohesive user experience. They enable users to access all the necessary tools and information in one place, improving efficiency and organization within the Salesforce Lightning Experience. This section outlines the process for creating a Lightning app tailored to a laptop rental CRM application.

What is a Lightning App?

A Lightning app is a collection of components within Salesforce that work together to serve a specific function. It includes:

- **Objects:** Relevant data objects that users interact with.
- **Tabs:** Navigation tabs that provide access to various components and records.
- **Utility Bar:** A customizable area for quick access to frequently used tools and features.
- **Branding:** Custom colours and logos to represent the app.

Lightning apps enhance productivity by allowing users to switch seamlessly between different apps and access all necessary tools from a unified interface.

Creating a Lightning App

To create a Lightning app for managing laptop rentals, follow these steps:

1. **Navigate to the Setup Page:**
 - Go to the Salesforce Setup page.
2. **Access the App Manager:**
 - In the Quick Find bar, type “App Manager” and select **App Manager** from the dropdown menu.
3. **Start Creating a New Lightning App:**
 - Click on **New Lightning App** to begin the app creation process.
4. **Configure App Details:**
 - **App Name:** Enter "LAPTOP RENTALS" in the App Name field.
 - Click **Next** to proceed to the next configuration page.
5. **Set App Options:**

- On the **App Options** page, keep the default settings as they are. These settings include the app's colour scheme and branding options.
- Click **Next** to move to the next section.

6. Configure Utility Items:

- On the **Utility Items** page, keep the default settings. This section allows you to add utility items like global actions or quick links.
- Click **Next** to proceed.

7. Upload App Photo:

- Upload a photo or icon that represents your app. This image will be used as the app's logo and will appear in the app launcher and navigation bar.

8. Add Navigation Items:

- **Select Navigation Items:** Use the search bar to find and select the following items: Total Laptops, Consumer, Laptop Booking, and Billing Process.
- Move these items to the selected list using the arrow button.
- Click **Next** to proceed to the user profile configuration.

9. Add User Profiles:

- **Search Profiles:** In the search bar, type "System Administrator" to find the relevant user profile.
- Select the profile and move it to the selected list using the arrow button.
- Click **Save & Finish** to finalize the app creation process.

Conclusion

Creating a Lightning app for your laptop rental CRM allows you to tailor the Salesforce experience to meet the specific needs of your business. By bundling relevant objects, tabs, and components into a single app, you create a streamlined and efficient interface for managing your laptop rental operations. The customization options for branding, utility items, and navigation ensure that users can work effectively and access all necessary tools and information with ease. Once created, your Lightning app will be available in the Salesforce navigation bar, ready to enhance productivity and organization within your organization.

App Settings

App Details & Branding

App Options

Utility Items (Desktop Only)

Navigation Items

User Profiles

App Details & Branding

Give your Lightning app a name and description. Upload an image and choose the highlight color for its navigation bar.

App Details

* App Name ⓘ

LAPTOP RENTALS

* Developer Name ⓘ

LAPTOP_RENTALS

Description ⓘ

Enter a description...

App Branding

Image ⓘ



Clear

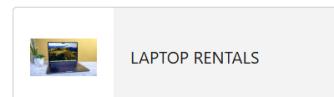
Primary Color Hex Value ⓘ

#0070D2

Org Theme Options

Use the app's image and color instead of the org's custom theme

App Launcher Preview



Topic 5: Fields in Salesforce

In Salesforce, fields represent the data stored within objects, similar to columns in a relational database. They are crucial for capturing, organizing, and managing data effectively. Fields enable streamlined searching, deletion, and editing of records.

Types of Fields

1. Standard Fields:

Standard fields are predefined by Salesforce and come with every object. These fields are designed to perform common tasks and are integral to Salesforce's functionality. Some key points about standard fields include:

- **Cannot be deleted:** Standard fields that are required by Salesforce cannot be deleted. Non-required standard fields can be deleted if not in use.
- **Common fields:** Typical examples include:
 - **Created By:** Tracks who created the record.
 - **Owner:** Indicates the record's owner.
 - **Last Modified:** Shows the last time the record was modified.
 - **Field Made During Object Creation:** Fields automatically created with the object.

2. Custom Fields:

Custom fields are user-defined fields that allow for greater flexibility and customization to meet specific organizational needs. Unlike standard fields, custom fields can be added, modified, or removed as needed. Users have full control over these fields, allowing them to tailor their Salesforce environment to their unique requirements.

A) Creating Fields in the Consumer Object:

1. Phone Number Field:

- Go to **Setup**.
- Click on **Object Manager**.
- Type "Consumer" in the search bar and select the object.
- Click on **Fields & Relationships**.
- Click **New**.
- Select **Data Type as Phone**.
- Click **Next**.

- Fill in the following details:
 - **Field Label:** Phone Number
 - **Field Name:** (auto-generated)
- Check the **Required** checkbox.
- Click **Next → Next → Save and New.**

2. Email Field:

- Go to **Setup**.
- Click on **Object Manager**.
- Type "Consumer" in the search bar and select the object.
- Click on **Fields & Relationships**.
- Click **New**.
- Select **Data Type** as **Email**.
- Click **Next**.
- Fill in the following details:
 - **Field Label:** Email
 - **Field Name:** (auto-generated)
- Click **Next → Next → Save and New.**

3. Address Field:

- Go to **Setup**.
- Click on **Object Manager**.
- Type "Consumer" in the search bar and select the object.
- Click on **Fields & Relationships**.
- Click **New**.
- Select **Data Type** as **Text Area**.
- Click **Next**.
- Fill in the following details:
 - **Field Label:** Address
 - **Field Name:** (auto-generated)

- Check the **Required** checkbox.
- Click **Next → Next → Save and New**.

4. Consumer Status Field:

- Go to **Setup**.
- Click on **Object Manager**.
- Type "Consumer" in the search bar and select the object.
- Click on **Fields & Relationships**.
- Click **New**.
- Select **Data Type** as **Picklist**.
- Click **Next**.
- Fill in the following details:
 - **Field Label:** Consumer Status
 - **Values:**
 - Student
 - Employee
 - Others (each on a new line)
- Click **Next → Next → Save and New**.

This documentation provides a foundational understanding of fields in Salesforce and detailed steps for creating fields in the Consumer object.

B) Adding a New Email Field to the Consumer Object

To add a new field to the Consumer object in Salesforce, follow these steps:

- 1. Navigate to Object Manager:**
 - Go to **Setup** in Salesforce.
 - Click on **Object Manager**.
- 2. Select the Object:**
 - In the search bar, type **Consumer**.
 - Click on the **Consumer** object from the list.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Email**.
- Click on **Next**.

5. Enter Field Details:

- **Field Label:** Email
- **Field Name:** This will be auto-generated based on the Field Label.

6. Set Field Attributes:

- Click **Next**.

7. Field Level Security:

- Configure field-level security as needed (default settings are usually fine for most cases).
- Click **Next**.

8. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save**.

9. Create Another Field (Optional):

- To create additional fields, click on **Save & New** to continue adding new fields.

This process allows you to add and configure an Email field for the Consumer object, ensuring it captures the relevant email information for each record.

C) Creating a Text Area Field in the Consumer Object

To add a Text Area field to the Consumer object in Salesforce, follow these steps:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Consumer**.
- Click on the **Consumer** object from the list.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Text Area**.
- Click on **Next**.

5. Enter Field Details:

- **Field Label:** Address
- **Field Name:** This will be auto-generated based on the Field Label.

6. Set Field Attributes:

- Select the **Required** checkbox to make this field mandatory.
- Click **Next**.

7. Field Level Security:

- Configure field-level security as needed (default settings are usually fine for most cases).
- Click **Next**.

8. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save**.

9. Create Another Field (Optional):

- To create additional fields, click on **Save & New** to continue adding new fields.

This process allows you to add and configure a Text Area field for the Consumer object, ensuring it captures detailed address information for each record.

D) Creating a Picklist Field in the Consumer Object

To add a Picklist field to the Consumer object in Salesforce, follow these steps:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Consumer**.
- Click on the **Consumer** object from the list.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Picklist**.
- Click on **Next**.

5. Enter Field Details:

- **Field Label:** Consumer Status
- **Field Name:** This will be auto-generated based on the Field Label.

6. Set Picklist Values:

- In the **Picklist Values** section, enter the values as follows, with each value on a new line:
 - Student
 - Employee
 - Others
- Select the **Required** checkbox to make this field mandatory.
- Click **Next**.

7. Field Level Security:

- Configure field-level security as needed (default settings are usually fine for most cases).
- Click **Next**.

8. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save**.

9. Create Another Field (Optional):

- To create additional fields, click on **Save & New** to continue adding new fields.

This process enables you to create a Picklist field for the Consumer object, allowing you to categorize consumers based on their status.

The screenshot shows the Salesforce Object Manager interface. The left sidebar has a 'Fields & Relationships' section selected. The main area displays a table titled 'Fields & Relationships' with 8 items, sorted by Field Label. The table columns are: FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The rows show the following fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text Area(255)		
consumer Status	consumer_Status__c	Picklist		
consumer_name	Name	Text(80)		✓
Created By	CreatedById	Lookup(User)		
Email	Email__c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone number	Phone_number__c	Phone		

E) Creating a Picklist Field in the Laptop Booking Object

To add a Picklist field to the Laptop Booking object in Salesforce, follow these steps:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Laptop Booking**.
- Click on the **Laptop Booking** object from the list.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Picklist**.
- Click on **Next**.

5. Enter Field Details:

- **Field Label:** (Enter a suitable label, such as "Laptop Brand")
- **Field Name:** This will be auto-generated based on the Field Label.

6. Set Picklist Values:

- In the **Picklist Values** section, enter the values as follows, with each value on a new line:
 - Dell
 - Acer
 - Hp
 - Mac
- Select the **Required** checkbox to make this field mandatory.
- Click **Next**.

7. Field Level Security:

- Configure field-level security as needed (default settings are usually fine for most cases).
- Click **Next**.

8. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save**.

9. Create Another Field (Optional):

- To create additional fields, click on **Save & New** to continue adding new fields.

This process will add a Picklist field to the Laptop Booking object, allowing users to select from predefined options for laptop brands.

F) Create a Picklist Field

To create a Picklist field in the Laptop Booking object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Laptop Booking**.
- Click on the **Laptop Booking** object.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Picklist**.
- Click on **Next**.

5. Enter Field Details:

- **Field Label:** (Enter a suitable label, such as "Processor Type")
- **Field Name:** This will be auto-generated based on the Field Label.

6. Set Picklist Values:

- In the **Picklist Values** section, enter the values as follows, with each value on a new line:
 - Core i3
 - Core i5
 - Core i7
- Select the **Required** checkbox to make this field mandatory.
- Click **Next**.

7. Field Level Security:

- Configure field-level security as needed (default settings are usually fine for most cases).
- Click **Next**.

8. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save**.

G) Create Field Dependency

To set up field dependency between two picklist fields in the Laptop Booking object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Laptop Booking**.
- Click on the **Laptop Booking** object.

3. Set Field Dependency:

- Click on **Fields & Relationships** on the left side.
- Click on **Field Dependencies**.

4. Create New Field Dependency:

- Click **New** to create a new field dependency.

5. Select Controlling and Dependent Fields:

- In the **Controlling Field** dropdown, select the picklist field that controls the dependency (e.g., "Laptop Brand").
- In the **Dependent Field** dropdown, select the picklist field that is affected by the dependency (e.g., "Processor Type").

6. Configure Picklist Values:

- Click on **Edit** to configure the picklist values for each controlling value.
- Check the boxes next to the processor types that should be available based on the selected laptop brand:
 - For **Dell**, include: Core i3, Core i5, Core i7.
 - For **Acer**, include: Core i3, Core i4, Core i5.
 - For **HP**, include: Core i3, Core i4, Core i5.
 - For **Mac**, include: (List specific processor types if applicable).

7. Save Configuration:

- Click **Save** to apply the field dependency settings.

This setup allows users to select different processor types based on the chosen laptop brand, ensuring that the available options are relevant and specific.

Fields & Relationships			
11 Items, Sorted by Field Label			
Created By	CreatedById	Lookup(User)	
Email	Email__c	Email	
how many months	how_many_months__c	Picklist	
Laptop Bookings	Name	Text(80)	✓
Laptop names	Laptop_type__c	Picklist	
Laptops Available	Laptops_Available__c	Formula (Number)	
Last Modified By	LastModifiedById	Lookup(User)	
Name	Name__c	Master-Detail(consumer)	✓
Total No Of Laptops	Total_Laptops__c	Master-Detail(Total Laptops)	✓

H) To create a Lookup Relationship field to the Consumer object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
 - Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Laptop Booking**.
 - Click on the **Laptop Booking** object.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
 - Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Lookup Relationship**.
 - Click on **Next**.

5. Select Related Object:

- In the **Related To** dropdown, select **Consumer**.
 - Click on **Next**.

6. Enter Field Details:

- **Field Label:** Change to **Name** (or another suitable label).
 - **Field Name:** This will be auto-generated based on the Field Label.

7. Configure Field-Level Security:

- Set the field-level security as needed (default settings are usually fine).
- Click **Next**.

8. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save**.

9. Create New:

- After saving, you can choose to create another field or return to the object manager.

This creates a Lookup Relationship field in the Laptop Booking object that links to the Consumer object, allowing users to associate each laptop booking with a specific consumer record.

I) Creating a Currency Field in the Laptop Booking Object

To create a Currency field in the Laptop Booking object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Laptop Booking**.
- Click on the **Laptop Booking** object.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Currency**.
- Click on **Next**.

5. Enter Field Details:

- **Field Label:** Enter Amount.

- **Length:** Enter **18** and **0** (for the number of decimal places).
- **Field Name:** This will be auto-generated based on the Field Label.

6. Configure Field-Level Security:

- Set the field-level security as needed (default settings are usually fine).
- Click **Next**.

7. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save**.

8. Create New:

- After saving, you can choose to create another field or return to the object manager.

J) Creating a Lookup Relationship Field in the Laptop Booking Object

To create a Lookup Relationship field in the Laptop Booking object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Laptop Booking**.
- Click on the **Laptop Booking** object.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Lookup Relationship**.
- Click on **Next**.

5. Specify the Related Object:

- In the **Related To** dropdown, select the object you want to relate to. For example, choose **consumer** if that is the related object.

- Click on **Next**.

6. Enter Field Details:

- **Field Label:** Enter the label for this field (e.g., **Name**).
- **Field Name:** This will be auto-generated based on the Field Label.

7. Configure Field-Level Security:

- Set the field-level security as needed (default settings are usually fine).
- Click **Next**.

8. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save**.

This will establish a lookup relationship between the **Laptop Booking** object and the related object (e.g., **consumer**), allowing you to reference records from the related object in your **Laptop Booking** records.

K) Creating a Lookup Relationship Field to the Total Laptops Object

To create a **Lookup Relationship** field in the **Laptop Booking** object that relates to the **Total Laptops** object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Laptop Booking**.
- Click on the **Laptop Booking** object.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Lookup Relationship**.
- Click on **Next**.

5. Specify the Related Object:

- In the **Related To** dropdown, select **Total Laptops**.
- Click on **Next**.

6. Enter Field Details:

- **Field Label:** Enter **Total No of Laptops**.
- **Field Name:** This will be auto-generated based on the Field Label.

7. Configure Field-Level Security:

- Set the field-level security as needed (default settings are usually fine).
- Click **Next**.

8. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save and New** if you want to create another field, or **save** if you're done.

This will create a lookup relationship field in the **Laptop Booking** object that references the **Total Laptops** object, allowing you to associate each laptop booking with a specific total laptop record.

The screenshot shows the Salesforce Object Manager for the 'Total Laptops' object. The left sidebar lists various setup options like Page Layouts, Lightning Record Pages, and Buttons. The main area is titled 'Fields & Relationships' and shows a table of fields. The table has columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The fields listed are: Created By (CreatedById, Lookup(User)), Laptops Available (Laptops_Available__c, Formula (Number)), Laptops delivered (Laptops_delivered__c, Roll-Up Summary (COUNT Laptop Bookings)), Last Modified By (LastModifiedById, Lookup(User)), Owner (OwnerId, Lookup(User,Group)), and Total Laptops (Name, Text(80)). The 'INDEXED' column shows checkmarks for Owner and Total Laptops.

Fields & Relationships					
6 Items, Sorted by Field Label					
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	
Created By	CreatedById	Lookup(User)			
Laptops Available	Laptops_Available__c	Formula (Number)			
Laptops delivered	Laptops_delivered__c	Roll-Up Summary (COUNT Laptop Bookings)			
Last Modified By	LastModifiedById	Lookup(User)			
Owner	OwnerId	Lookup(User,Group)		✓	
Total Laptops	Name	Text(80)		✓	

L) Creating an Email Field in the Laptop Booking Object

To create an **Email** field in the **Laptop Booking** object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Object:

- In the search bar, type **Laptop Booking**.
- Click on the **Laptop Booking** object.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Email**.
- Click on **Next**.

5. Enter Field Details:

- **Field Label:** Enter the appropriate label for the field.
- **Field Name:** This will be auto-generated based on the Field Label.

6. Configure Field-Level Security:

- Set the field-level security as needed (default settings are usually fine).
- Click **Next**.

7. Add to Page Layouts:

- Select the page layouts where this field should be included.
- Click **Save**.

NOTE: Ensure that you fill in the records you created in both the **Consumer** and **Laptop Bookings** objects and establish the necessary relationships between them. After saving the records, go to the **Laptop Bookings** object and edit the lookup relationships to set up the detailed relationship.

M) Creating a Roll-up Summary Field in the Total Laptops Object

To create a Roll-up Summary field in the Total Laptops object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Total Laptops Object:

- In the search bar, type **Total Laptops**.
- Click on the **Total Laptops** object.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Roll-up Summary**.
- Click **Next**.

5. Enter Field Details:

- **Field Label:** Enter **Laptops Delivered**.
- **Field Name:** This will be auto-generated based on the Field Label.

6. Configure Roll-up Summary:

- **Summarized Object:** Select **Laptop Bookings**.
- **Roll-up Type:** Select **COUNT**.
- Click **Next**.

7. Complete the Setup:

- Review the details and click **Save**.

N) Creating a Formula Field in the Laptop Booking Object

To create a Formula field in the Laptop Booking object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Laptop Booking Object:

- In the search bar, type **Laptop Booking**.
- Click on the **Laptop Booking** object.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.

- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Formula**.
- Click **Next**.

5. Enter Field Details:

- **Field Label:** Enter **Laptops Available**.
- **Field Name:** This will be auto-generated based on the Field Label.
- **Formula Return Type:** Select **Number**.
- **Decimal Places:** Select **0**.
- Click **Next**.

6. Configure Formula:

- Click on **Advanced Formula**.
- Enter the formula: `50 - Total_no_of_laptops__r.Laptops_delivered__c`.
- Click on **Insert Field**, then select the **Total No Of Laptops** field and **Laptops Delivered** field from the pop-up window.
- Click on **Check Syntax** to ensure there are no errors.

7. Complete the Setup:

- Click **Next**.
- Review the details and click **Save and New**.

O) Creating a Picklist Field in the Laptop Booking Object

To create a Picklist field in the Laptop Booking object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Laptop Booking Object:

- In the search bar, type **Laptop Booking**.
- Click on the **Laptop Booking** object.

3. Create a New Field:

- Click on **Fields & Relationships** on the left side.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Picklist**.
- Click **Next**.

5. Enter Field Details:

- **Picklist Values:** Enter values 1, 2, 3, 4, 5 separated by new lines.
- Click **Next**.

6. Complete the Setup:

- Review the details and click **Save**.

P) Creation of Fields & Relationships for the Billing Process Object

1. Creating a Master-Detail Relationship Field

To create a Master-Detail Relationship field in the Billing Process object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Billing Process Object:

- Type **Billing Process** in the search bar.
- Click on the **Billing Process** object.

3. Create a New Field:

- Click on **Fields & Relationships**.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Master-Detail Relationship**.
- Click **Next**.

5. Configure Relationship:

- **Related To:** Select **Consumer** from the drop-down list.
- Click **Next**.

6. Enter Field Details:

- **Field Label:** Enter **Name**.
- **Field Name:** This will be auto-generated.

7. Complete the Setup:

- Click **Next**.
- Click **Next** again.
- Click **Save and New**.

2. Creating a Lookup Relationship Field

To create a **Lookup Relationship** field in the **Billing Process** object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Billing Process Object:

- Type **Billing Process** in the search bar.
- Click on the **Billing Process** object.

3. Create a New Field:

- Click on **Fields & Relationships**.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Lookup Relationship**.
- Click **Next**.

5. Configure Relationship:

- **Related To:** Select **Laptop Booking** from the drop-down list.
- Click **Next**.

6. Enter Field Details:

- **Field Label:** Enter **Laptop Booking**.

- **Field Name:** This will be auto-generated.

7. Complete the Setup:

- Click **Next**.
- Click **Next** again.
- Click **Save and New**.

3. Creating a Picklist Field

To create a Picklist field in the Billing Process object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Billing Process Object:

- Type **Billing Process** in the search bar.
- Click on the **Billing Process** object.

3. Create a New Field:

- Click on **Fields & Relationships**.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Picklist**.
- Click **Next**.

5. Enter Field Details:

- **Field Label:** Enter **Payment Mode**.
- **Picklist Values:** Enter values separated by new lines:
 - Cash
 - Check
 - Credit card
 - Debit card
 - UPI
 - Phonepe

- Gpay
- Paytm
- Select **Required**.

6. Complete the Setup:

- Click **Next**.
- Click **Next** again.
- Click **Save and New**.

4. Creating a Cross-Object Formula Field

To create a Cross-Object Formula field in the Billing Process object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Billing Process Object:

- Type **Billing Process** in the search bar.
- Click on the **Billing Process** object.

3. Create a New Field:

- Click on **Fields & Relationships**.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Formula**.
- Click **Next**.

5. Enter Field Details:

- **Field Label:** Enter Amount.
- **Field Name:** This will be auto-generated.
- **Formula Return Type:** Select Number.
- Click **Next**.

6. Configure Formula:

- Click on **Advanced Formula**.

- Click on **Insert Field**.
- In the pop-up window, select **Billing Process** and then **Laptop Booking**.
- Select **Amount** from the third drop-down.
- The formula will be: Laptop_Booking__r.Amount__c.
- Click **Insert**.
- Click **Check Syntax** to ensure there are no errors.

7. Complete the Setup:

- Click **Next**.
- Click **Next** again.
- Click **Save and New**.

Fields & Relationships 7 Items, Sorted by Field Label				
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Amount	Amount__c	Formula (Number)		
Billing ProcessName	Name	Text(80)		✓
Created By	CreatedBy	Lookup(User)		
Laptop Bookings	Laptop_Bookings__c	Lookup(Laptop Bookings)		✓
Last Modified By	LastModifiedBy	Lookup(User)		
Name	Name__c	Master-Detail(consumer)		✓
Payment Mode	Payment_Mode__c	Picklist		

Creating a Formula Field in the Total Laptops Object

To create a **Formula field** in the **Total Laptops** object:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the **Total Laptops** Object:

- Type **Total Laptops** in the search bar.
- Click on the **Total Laptops** object.

3. Create a New Field:

- Click on **Fields & Relationships**.
- Click on the **New** button.

4. Define Field Type:

- Select **Data Type** as **Formula**.
- Click **Next**.

5. Enter Field Details:

- **Field Label:** Enter **Laptops Available**.
- **Field Name:** This will be auto-generated.
- **Formula Return Type:** Select **Number**.
- **Decimal Places:** Select **0**.
- Click **Next**.

6. Configure Formula:

- Click on **Advanced Formula**.
- In the formula box, enter: `50 - Laptops_delivered__c`.
- **Note:** Here, `Laptops_delivered__c` should be replaced with the actual API name of the field in the Total Laptops object.
- Click **Check Syntax** to ensure there are no errors.

7. Complete the Setup:

- Click **Next**.
- Click **Next** again.
- Click **Save and New**.

This formula calculates the number of laptops available by subtracting the number of laptops delivered from a total of 50.

Total Laptops

Fields & Relationships		Quick Find		New	Deleted Fields	Field Dependencies	Set History Tracking
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD		INDEXED		
Created By	CreatedById	Lookup(User)					
Laptops Available	Laptops_Available__c	Formula (Number)					
Laptops delivered	Laptops_delivered__c	Roll-Up Summary (COUNT Laptop Bookings)					
Last Modified By	LastModifiedById	Lookup(User)					
Owner	OwnerId	Lookup(User,Group)			✓		
Total Laptops	Name	Text(80)			✓		

Topic 6: Creating a Validation Rule for Phone Number and Email in the Consumer Object

To ensure that both the phone number and email fields are not left blank in the Consumer object, follow these steps:

1. Navigate to Object Manager:

- Go to **Setup** in Salesforce.
- Click on **Object Manager**.

2. Select the Consumer Object:

- From the list, find and click on the **Consumer** object.
- Click **Edit** from the dropdown menu if necessary.

3. Access Validation Rules:

- Click on **Validation Rules** in the left-hand menu.
- Click on the **New** button.

4. Define the Validation Rule:

- **Rule Name:** Enter Phonenumberoremailblankrule.
- **Description:** Enter Phone number and email should not be blank.

5. Enter the Formula:

- **Formula:** Enter the following formula to check if either the phone number or email field is blank:

OR(

ISBLANK(phone_number__c),

ISBLANK(email__c)

)

- Click **Check Syntax** to ensure the formula is correct.

6. Specify the Error Message:

- Add an appropriate error message to be displayed when the validation rule is triggered.

7. Save the Validation Rule:

- Click **Save** to create the validation rule.

Topic 7: Profiles in Salesforce

A profile in Salesforce is a collection of settings and permissions that define what a user can and cannot do. Profiles control various aspects such as object permissions, field permissions, user permissions, tab settings, app settings, and more. They help in defining the roles and responsibilities of users based on their job functions, such as System Administrator, Developer, or Sales Representative.

Types of Profiles in Salesforce

1. Standard Profiles:

- **Contract Manager**
- **Read Only**
- **Marketing User**
- **Solutions Manager**
- **Standard User**
- **System Administrator**

Standard profiles come with predefined permissions for standard objects and cannot be deleted.

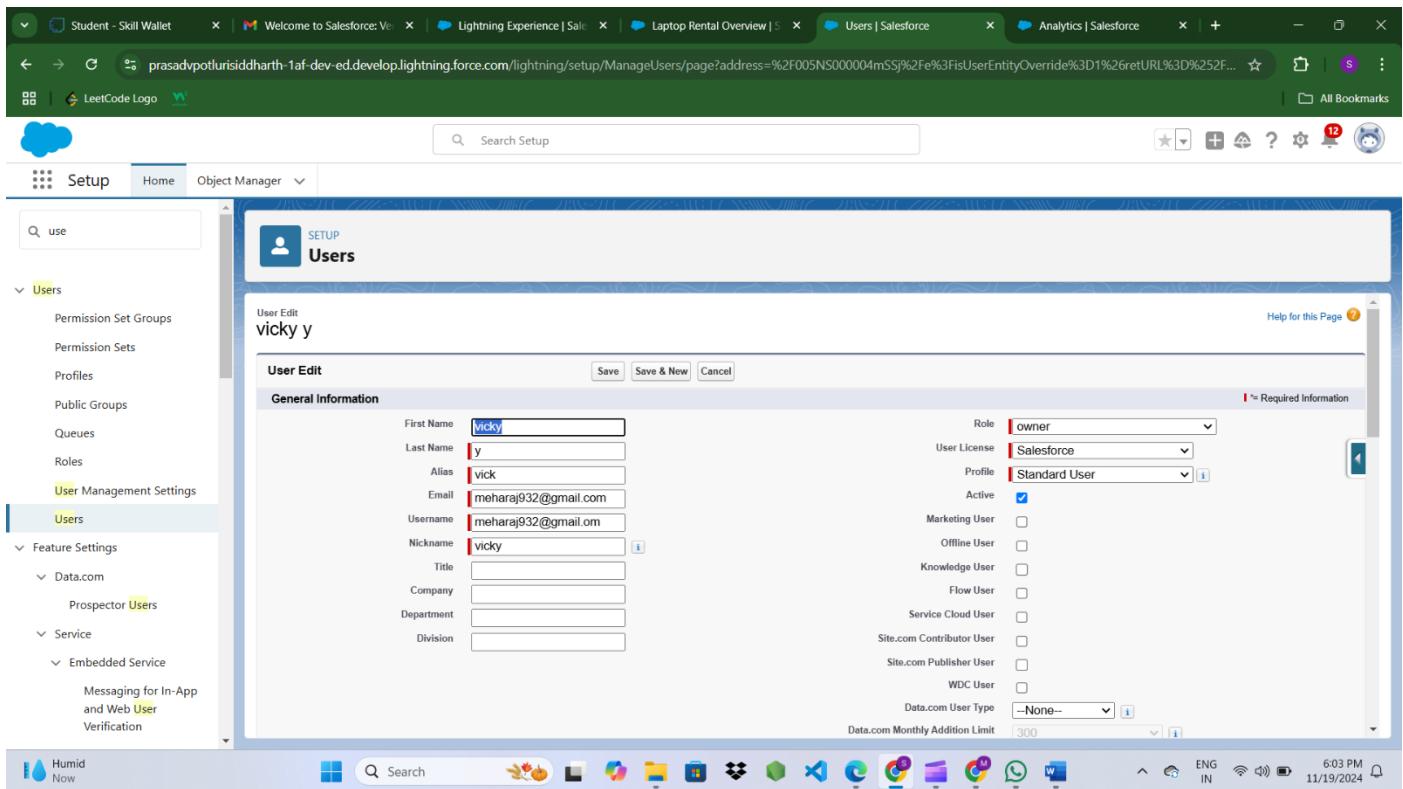
2. Custom Profiles:

- Created and defined by users based on specific needs.
- Can be deleted if no users are assigned to them.

Creating and Configuring Profiles

Owner Profile:

1. Go to Setup.
2. Type **Profiles** in the Quick Find box and click on **Profiles**.
3. Clone the **Standard User** profile.
4. Enter **Profile Name** as **Owner** and click **Save**.
5. Scroll down to **Custom Object Permissions**.
6. Provide access permissions for the following objects: **Total Laptops**, **Consumers**, **Laptop Booking**, and **Billing Process** as required.
7. Click **Save**.



Agent Profile:

1. Go to Setup.
2. Type **Profiles** in the Quick Find box and click on **Profiles**.
3. Clone the **Standard Platform User** profile.
4. Enter **Profile Name** as **Agent** and click **Save**.
5. Click **Edit** on the Agent profile page.
6. Scroll down to **Custom Object Permissions**.
7. Provide access permissions for the following objects: **Total Laptops**, **Consumer**, **Laptop Bookings**, and **Billing Process** as required.
8. Click **Save**.

This process ensures that users with different roles have the appropriate permissions to access and interact with the necessary objects and functionalities in Salesforce.

The screenshot shows the Salesforce Setup interface. The left sidebar is titled 'Setup' and includes sections for Home, Object Manager, and various system settings. The 'Users' section is currently selected. The main content area is titled 'User Edit' and shows a user record for 'ram ram'. The 'General Information' section contains the following fields:

Field	Value
First Name	ram
Last Name	ram
Alias	ramu
Email	meharaj123@gmail.com
Username	meharaj123@gmail.com
Nickname	ram
Title	(empty)
Company	(empty)
Department	(empty)
Division	(empty)
Role	Agent
User License	Salesforce Platform
Profile	Standard Platform User
Active	<input checked="" type="checkbox"/>
Marketing User	<input type="checkbox"/>
Offline User	<input type="checkbox"/>
Knowledge User	<input type="checkbox"/>
Flow User	<input type="checkbox"/>
Service Cloud User	<input type="checkbox"/>
Site.com Contributor User	<input type="checkbox"/>
Site.com Publisher User	<input type="checkbox"/>
WDC User	<input type="checkbox"/>
Data.com User Type	—None—
Data.com Monthly Addition Limit	300

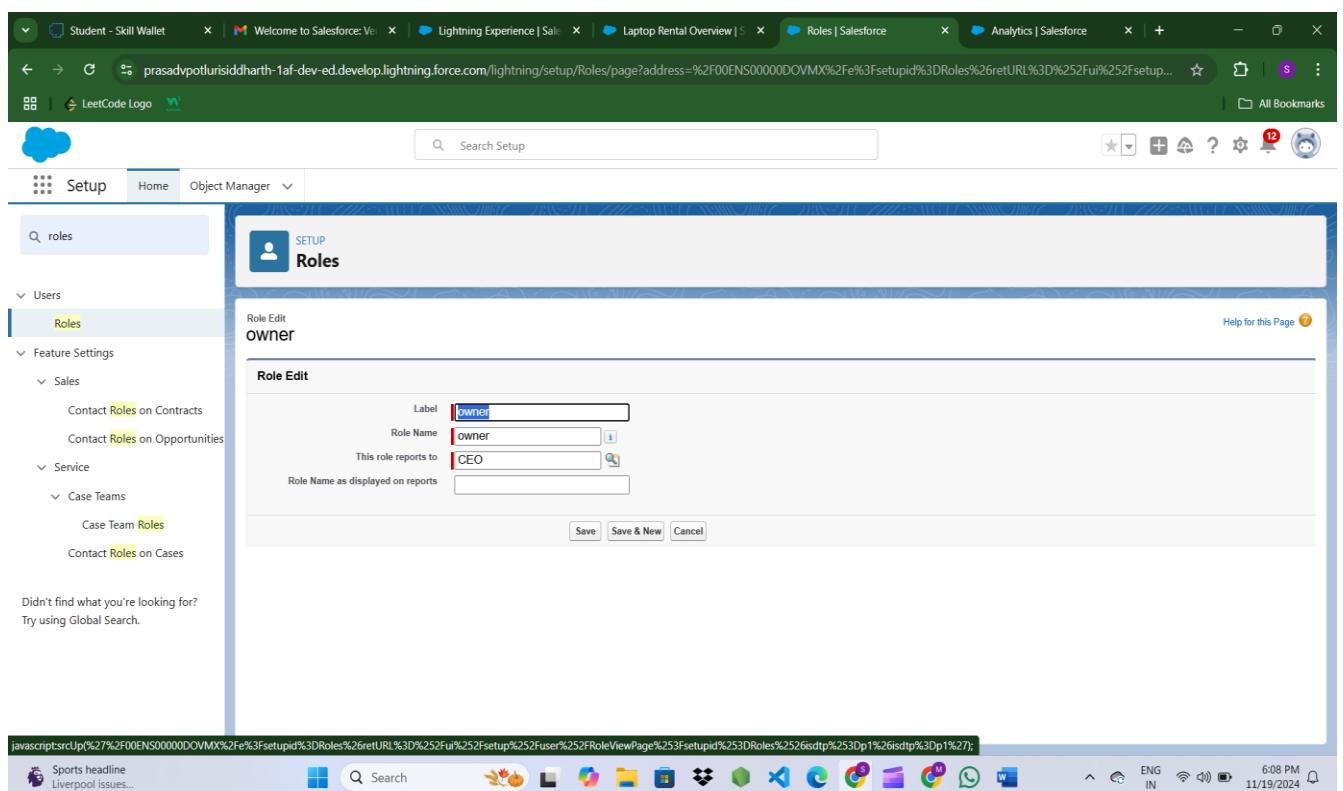
At the bottom of the page, there are standard browser controls for back, forward, search, and refresh, along with a status bar showing the date and time.

Topic 8: Roles and Hierarchy in Salesforce

Roles in Salesforce determine a user's visibility and access to records within the organization. They define what users can see and access at the record level based on their role in the organizational hierarchy.

Creating an Owner Role

1. Go to Setup.
2. In the Quick Find box, type **Roles** and click on **Set Up Roles**.
3. Click **Expand All** to view the existing roles.
4. Click **Add Role** under the role where this new role will be positioned.
5. Enter **Label as Owner**. The **Role Name** will auto-populate.
6. Click **Save** to create the Owner role.



7.

Creating Agent Roles

1. Go to Setup.
2. In the Quick Find box, type **Roles** and click on **Set Up Roles**.
3. Click the plus (+) sign next to the **CEO** role, and select **Add Role** under the **Owner** role.

4. Enter **Label** as Agent. The **Role Name** will auto-populate.

5. Click **Save** to create the Agent role.

This setup allows you to define the hierarchy within your Salesforce organization, ensuring that users have the appropriate level of access based on their role.

The screenshot shows the Salesforce Setup Roles page. On the left, the navigation sidebar is open, showing 'Users' and 'Roles' under 'Sales'. In the main content area, a 'Role Edit' form is displayed for a role named 'Agent'. The 'Label' field contains 'Agent', and the 'Role Name' field also contains 'Agent'. The 'This role reports to' field has 'owner' selected. Below the form are 'Save', 'Save & New', and 'Cancel' buttons. The top navigation bar includes tabs for 'Student - Skill Wallet', 'Welcome to Salesforce', 'Lightning Experience | Sale', 'Laptop Rental Overview | S', 'Roles | Salesforce', 'Analytics | Salesforce', and 'All Bookmarks'. The status bar at the bottom shows 'Sports headline NBA star fined \$...', 'Search', and system status like 'ENG IN' and '6:09 PM 11/19/2024'.

The screenshot shows the Salesforce Setup Roles page with the 'Roles' tab selected in the sidebar. The main content area displays the 'Creating the Role Hierarchy' section. It shows a tree view of roles under 'Your Organization's Role Hierarchy'. At the top level is 'Prasad v Poturi Siddhartha Institute of Technology', which has a child node 'CEO'. Below 'CEO' is another 'Add Role' node. A 'Show in tree view' button is located in the top right corner of this section. The top navigation bar and status bar are identical to the previous screenshot.

Topic 9: Creating Users in Salesforce

Users in Salesforce are individuals who need access to the platform, such as sales reps, managers, or IT specialists. Each user has a specific account with settings that determine their access and features available to them.

Creating a User

1. Navigate to User Setup:

- Go to Setup.
- In the Quick Find box, type **Users**.
- Select **Users** from the search results.

2. Create a New User:

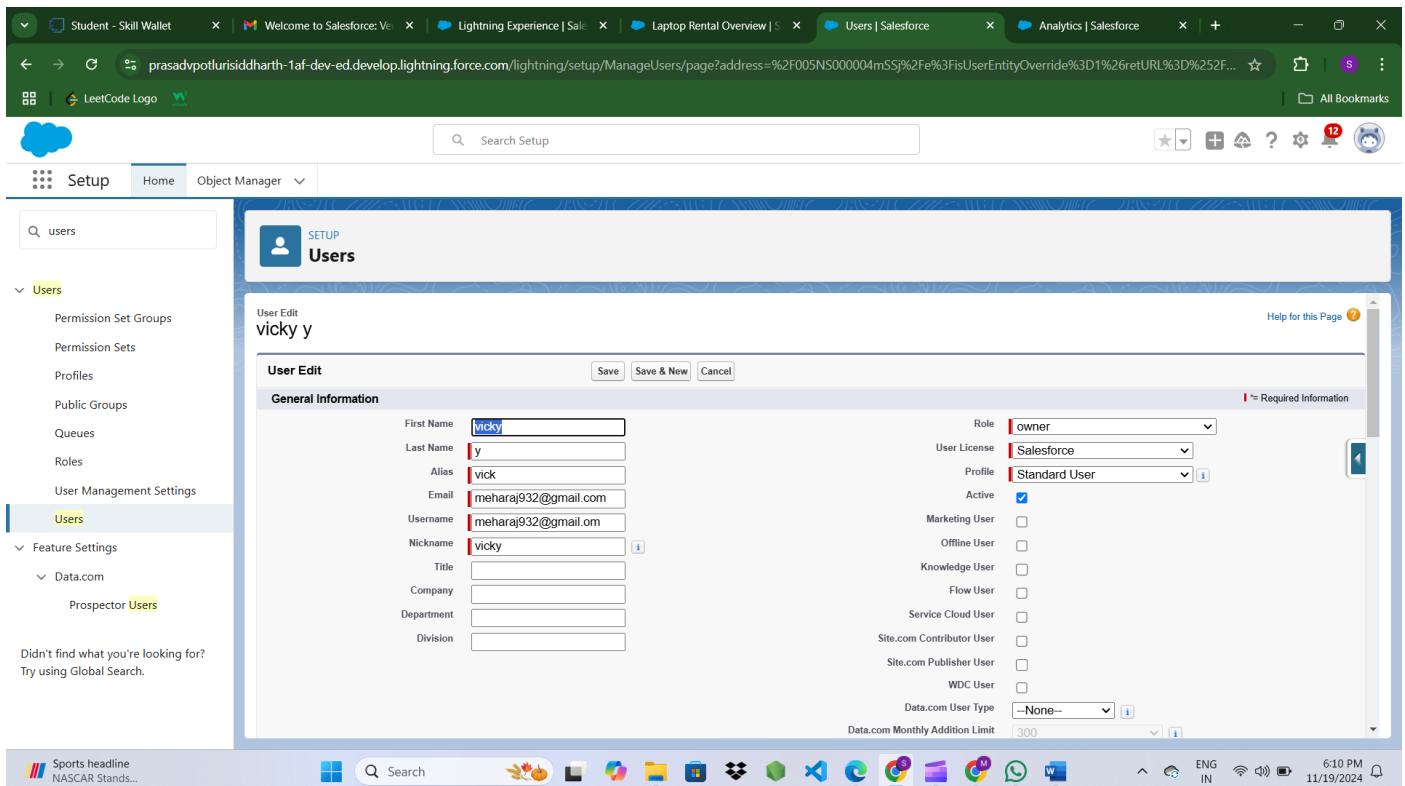
- Click **New User**.

3. Fill in the Required Fields:

- **First Name:** Vicky
- **Last Name:** Y
- **Alias:** Enter an alias (e.g., vicky.y)
- **Email:** Enter your personal email address.
- **Username:** Should be in the format text@text.text (e.g., vicky.y@company.com).
- **Nickname:** Enter a nickname (e.g., Vicky)
- **Role:** Select **Owner**.
- **User License:** Select **Salesforce**.
- **Profile:** Select **Owner**.

4. Save the User:

- Click **Save**.



Creating Another User

1. Navigate to User Setup:

- Go to Setup.
- In the Quick Find box, type **Users**.
- Select **Users** from the search results.

2. Create a New User:

- Click **New User**.

3. Fill in the Required Fields:

- **First Name:** Ram
- **Last Name:** Ram
- **Alias:** Enter an alias (e.g., ram.ram)
- **Email:** Enter your personal email address.
- **Username:** Should be in the format text@text.text (e.g., ram.ram@company.com).
- **Nickname:** Enter a nickname (e.g., Ram)
- **Role:** Select **Agent**.
- **User License:** Select **Salesforce Platform**.

- **Profile:** Select **Standard Platform User**.

4. Save the User:

- Click **Save**

The screenshot shows the Salesforce Setup interface with the 'Users' page open. The user 'ram' is being edited. The 'General Information' section contains the following data:

Field	Value
First Name	ram
Last Name	ram
Alias	ramu
Email	meharaj123@gmail.com
Username	meharaj123@gmail.com
Nickname	ram
Title	(empty)
Company	(empty)
Department	(empty)
Division	(empty)
Role	Agent
User License	Salesforce Platform
Profile	Standard Platform User
Active	<input checked="" type="checkbox"/>
Marketing User	<input type="checkbox"/>
Offline User	<input type="checkbox"/>
Knowledge User	<input type="checkbox"/>
Flow User	<input type="checkbox"/>
Service Cloud User	<input type="checkbox"/>
Site.com Contributor User	<input type="checkbox"/>
Site.com Publisher User	<input type="checkbox"/>
WDC User	<input type="checkbox"/>
Data.com User Type	None

By creating users and assigning them appropriate roles, licenses, and profiles, you can control what each individual can access and manage within your Salesforce environment.

Topic 10: Creating a Flow in Salesforce

Introduction to Flows

In Salesforce, flows are a powerful tool for automating business processes, collecting and updating data, and guiding users through a series of screens or steps. They use a visual interface and can be created without coding knowledge. Here are some key types of flows:

- **Screen Flows:** Guide users through a series of screens for data entry or display.
- **Autolaunched Flows:** Triggered by events (e.g., record creation/update) and do not require user interaction.
- **Scheduled Flows:** Run at specific times or intervals for recurring tasks.
- **Record-Triggered Flows:** Triggered when records meet specified criteria.
- **Flow Builder:** Visual interface for designing flows.
- **Flow Templates:** Pre-built templates for various use cases.
- **Flow Elements:** Components like variables, decisions, and loops used to build flows.
- **Subflows:** Reusable flow elements for managing complex processes.

Why Create a Flow?

Flows are essential for automating complex processes and enhancing efficiency in Salesforce. For example, creating a flow can automatically generate amounts based on the selected laptop type.

Steps to Create a Flow

Objective: To automatically populate the Amount field based on the selected laptop type.

1. Navigate to Flow Builder:

- Go to **Setup**.
- Type "**Flows**" in the Quick Find box.
- Click on **Flows**.
- Click **New Flow**.

2. Select Flow Type:

- Choose **Autolaunched Flow** if the flow will run in the background without user interaction.
- Alternatively, select **Screen Flow** if user interaction is needed.

3. Add Flow Elements:

- Drag and drop the necessary elements (e.g., **Get Records** to retrieve laptop type data, **Update Records** to set the amount).
- Use **Decisions** to handle logic based on the selected laptop type.

4. Configure Elements:

- **Get Records:** Retrieve laptop type data from the database.
- **Decision:** Determine the amount based on the laptop type.
- **Update Records:** Set the Amount field based on the decision outcome.

5. Test the Flow:

- Use the **Debug** feature to test the flow and ensure it works as expected.
- Make adjustments as needed based on test results.

6. Activate the Flow:

- Click **Save** to save your work.
- Click **Activate** to make the flow live and operational.

7. Deploy the Flow:

- Ensure that the flow is properly deployed and integrated into your Salesforce environment.

By creating this flow, you can automate the process of setting amounts based on laptop types, improving efficiency and accuracy in your Salesforce application.

1) Creating a Flow for Dell Laptop in Salesforce

Objective: To create a Record-Triggered Flow that calculates and updates the amount based on the selected Dell laptop type and the number of months for the laptop booking.

Steps to Create the Flow:

1. Navigate to Flow Builder:

- Go to **Setup**.
- Type "**Flows**" in the Quick Find box.
- Click on **Flows**.
- Click **New Flow**.

2. Select Flow Type:

- Choose **Record-Triggered Flow**.

- Click **Create**.

3. Configure Flow Trigger:

- **Object:** Select **Laptop Booking** from the drop-down list.
- **Trigger Flow When:** Choose **A record is Created or Updated**.
- **Optimize the Flow for:** Select **Actions and Related Records**.
- Click **Done**.

4. Add and Configure Decision Elements:

First Decision Element:

- Click on the + symbol.
- Select **Decision**.
- **Details Label:** Enter **Laptop Type Check**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **dell**.
- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.Laptop Booking**.
- **Operator:** Select **Equals**.
- **Value:** Enter **dell**.
- Click **Done**.

Repeat the above step for other laptop types: **acer, hp, mac**.

Edit Decision

* Label field should updated	* API Name field_should_updated																					
Description the field should be automatically updated																						
Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.																						
OUTCOME ORDER 0 + OUTCOME DETAILS <div style="float: right;">Delete Outcome</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%;">dell</td> <td>* Label dell</td> <td>* Outcome API Name dell</td> </tr> <tr> <td>acer</td> <td colspan="2"></td> </tr> <tr> <td>hp</td> <td colspan="2"></td> </tr> <tr> <td>mac</td> <td colspan="2"></td> </tr> <tr> <td>false</td> <td colspan="2"> Condition Requirements to Execute Outcome All Conditions Are Met (AND) </td> </tr> <tr> <td></td> <td>Resource \$Record > Laptop names</td> <td>Operator Equals</td> </tr> <tr> <td></td> <td></td> <td>Value Dell</td> </tr> </table> <div style="clear: both; margin-top: 10px;"> Cancel Done </div>		dell	* Label dell	* Outcome API Name dell	acer			hp			mac			false	Condition Requirements to Execute Outcome All Conditions Are Met (AND)			Resource \$Record > Laptop names	Operator Equals			Value Dell
dell	* Label dell	* Outcome API Name dell																				
acer																						
hp																						
mac																						
false	Condition Requirements to Execute Outcome All Conditions Are Met (AND)																					
	Resource \$Record > Laptop names	Operator Equals																				
		Value Dell																				

Dell Sub-Decision Element:

- Click on the + symbol beside **dell**.
- Select **Decision**.
- **Details Label:** Enter **Core Type Check**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **dell core i3**.
- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.core type**.
- **Operator:** Select **Equals**.
- **Value:** Enter **core i3**.
- Click **Done**.

Repeat for **core i5** and **core i7**.

Dell Core i3 Sub-Decision Element:

- Click on the + symbol beside **core i3**.
- Select **Decision**.
- **Details Label:** Enter **Months Selected**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **dell 1(i3)**.

- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.how many months**.
- **Operator:** Select **Equals**.
- **Value:** Enter **1**.
- **Click Done.**

Repeat for **dell 2(i3)**, **dell 3(i3)**, **dell 4(i3)**, **dell 5(i3)** with corresponding values **2, 3, 4, 5**.

Dell Core i7 Sub-Decision Element:

- Click on the + symbol beside **core i7**.
- Select **Decision**.
- **Details Label:** Enter **Months Selected**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **dell 1(i7)**.
- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.how many months**.
- **Operator:** Select **Equals**.
- **Value:** Enter **1**.
- **Click Done.**

Repeat for **dell 2(i7)**, **dell 3(i7)**, **dell 4(i7)**, **dell 5(i7)** with corresponding values **2, 3, 4, 5**.

The screenshot shows the 'Edit Decision' dialog box. At the top, there are fields for 'Label' (months selected) and 'API Name' (months_selected). Below these are 'Description' and 'Outcomes' sections. The 'Outcomes' section contains an 'OUTCOME ORDER' table with five rows labeled 1 through 5. Each row has a 'Label' field (e.g., 1, X1), an 'Outcome API Name' field (e.g., X1), and a 'Condition Requirements to Execute Outcome' dropdown set to 'All Conditions Are Met (AND)'. Row 1 also includes a 'Resource' field with '\$Record > how many months' and an 'Operator' field with 'Equals' and a 'Value' of '1'. At the bottom right are 'Cancel' and 'Done' buttons.

5. Add Update Records Actions:

For Dell Core i3:

- Click on the + symbol after **dell 1(i3)**, **dell 2(i3)**, etc.
- Select **Update Records**.
- **Details Label:** Enter **Update Amount for Dell i3**.
- **API Name:** Automatically generated.
- **Field:** Select **Amount__c**.
- **Value:** Set corresponding values for **dell 1(i3)** to **1000**, **dell 2(i3)** to **2000**, and so on.
- Click **Done**.

For Dell Core i7:

- Click on the + symbol after **dell 1(i7)**, **dell 2(i7)**, etc.
- Select **Update Records**.
- **Details Label:** Enter **Update Amount for Dell i7**.
- **API Name:** Automatically generated.
- **Field:** Select **Amount__c**.
- **Value:** Set corresponding values for **dell 1(i7)** to **2000**, **dell 2(i7)** to **4000**, and so on.
- Click **Done**.

Edit Decision

* Label	field updated	* API Name	field_updated			
Description						
Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.						
OUTCOME ORDER	+ dell core i3 dell core i5 dell core i7	OUTCOME DETAILS	Delete Outcome			
		* Label dell core i3	* Outcome API Name dellcore_i3			
		Condition Requirements to Execute Outcome All Conditions Are Met (AND)				
Default Outcome	<table border="1"><tr><td>Resource \$Record > core type</td><td>Operator Equals</td><td>Value core i3</td></tr></table>			Resource \$Record > core type	Operator Equals	Value core i3
Resource \$Record > core type	Operator Equals	Value core i3				
	Cancel Done					

2) Creating a Flow for Acer Laptop in Salesforce

Objective: To create a Record-Triggered Flow that calculates and updates the amount based on the selected Acer laptop type and the number of months for the laptop booking.

Steps to Create the Flow:

1. Navigate to Flow Builder:

- Go to **Setup**.
- Type "**Flows**" in the Quick Find box.
- Click on **Flows**.
- Click **New Flow**.

2. Select Flow Type:

- Choose **Record-Triggered Flow**.
- Click **Create**.

3. Configure Flow Trigger:

- **Object:** Select **Laptop Booking** from the drop-down list.
- **Trigger Flow When:** Choose **A record is Created or Updated**.
- **Optimize the Flow for:** Select **Actions and Related Records**.
- Click **Done**.

4. Add and Configure Decision Elements:

The screenshot shows the 'Edit Decision' interface. At the top, there's a header 'Edit Decision'. Below it, a section titled 'Outcomes' with the sub-instruction: 'For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.' A table lists three outcomes:

OUTCOME ORDER	OUTCOME DETAILS	DELETE OUTCOME
1 acer core i3	*Label: acer core i3 *Outcome API Name: acer_core_i3	Delete Outcome
2 acer core i5		
3 acer core i7		

Below the table, under 'Default Outcome', there's a condition setup with fields: 'Resource' (\$Record > core type), 'Operator' (Equals), and 'Value' (core i3). A button '+ Add Condition' is available. Under 'When to Execute Outcome', two radio buttons are shown: 'If the condition requirements are met' (selected) and 'Only if the record that triggered the flow to run is updated to meet the condition requirements'. At the bottom right are 'Cancel' and 'Done' buttons.

5.

Decision for Acer Laptop:

- Click on the + symbol next to **acer**.
- Select **Decision**.
- **Details Label:** Enter **Core Type Check**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **acer core i3**.
- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.core type**.
- **Operator:** Select **Equals**.
- **Value:** Enter **core i3**.
- Click **Done**.

Acer Core i3 Sub-Decision Element:

- Click on the + symbol beside **acer core i3**.
- Select **Decision**.
- **Details Label:** Enter **Months Selected**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **acer 1(i3)**.
- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.how many months**.
- **Operator:** Select **Equals**.
- **Value:** Enter **1**.
- Click **Done**.

Repeat for **acer 2(i3)**, **acer 3(i3)**, **acer 4(i3)**, **acer 5(i3)** with corresponding values **2, 3, 4, 5**.

Edit Decision

*Label acer months selected	*API Name acer_months_selected
Description 	
Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.	
OUTCOME ORDER ① + <ul style="list-style-type: none"> acer 1(i3) acer 2(i3) acer 3(i3) acer 4(i3) acer 5(i3) 	OUTCOME DETAILS <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> *Label acer 1(i3) *Outcome API Name acer_1_i3 </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> Condition Requirements to Execute Outcome All Conditions Are Met (AND) </div> <div style="display: flex; justify-content: space-between;"> Resource Operator Value </div> <div style="display: flex; align-items: center;"> \$Record > how many months Equals 1 Save </div>
Cancel Done	

6. Add Update Records Actions:

For Acer Core i3:

- Click on the + symbol after **acer 1(i3)**, **acer 2(i3)**, etc.
- Select **Update Records**.
- **Details Label:** Enter **Update Amount for Acer i3**.
- **API Name:** Automatically generated.
- **Field:** Select **Amount__c**.
- **Value:** Set corresponding values:
 - **acer 1(i3):** 900
 - **acer 2(i3):** 1800
 - **acer 3(i3):** 2700
 - **acer 4(i3):** 3600
 - **acer 5(i3):** 4800
- Click **Done**.

Edit Update Records

one month of acer i3 rate (one_month_of_acer_i3_rate) 

*How to Find Records to Update and Set Their Values

Use the laptop bookings record that triggered the flow
 Update records related to the laptop bookings record that triggered the flow
 Use the IDs and all field values from a record or record collection
 Specify conditions to identify records, and set fields individually

Set Filter Conditions

Condition Requirements to Update Record

None—Always Update Record 

Set Field Values for the Laptop Bookings Record

Field	Value	
Amount_c	900	

3) Creating a Flow for HP Laptop in Salesforce

Objective: To create a Record-Triggered Flow that calculates and updates the amount based on the selected HP laptop type and the number of months for the laptop booking.

Steps to Create the Flow:

1. Navigate to Flow Builder:

- Go to **Setup**.
- Type "**Flows**" in the Quick Find box.
- Click on **Flows**.
- Click **New Flow**.

2. Select Flow Type:

- Choose **Record-Triggered Flow**.
- Click **Create**.

3. Configure Flow Trigger:

- **Object:** Select **Laptop Booking** from the drop-down list.
- **Trigger Flow When:** Choose **A record is Created or Updated**.
- **Optimize the Flow for:** Select **Actions and Related Records**.
- Click **Done**.

4. Add and Configure Decision Elements:

Decision for HP Laptop:

- Click on the + symbol next to **hp**.
- Select **Decision**.
- **Details Label:** Enter **Core Type Check**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **hp core i5**.
- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.core type**.
- **Operator:** Select **Equals**.
- **Value:** Enter **hp i5**.
- Click **Done**.

The screenshot shows the 'Edit Decision' screen. At the top, it says 'Outcomes: For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.' Below this, there's a table-like structure for defining outcomes:

OUTCOME ORDER	OUTCOME DETAILS	Condition Requirements to Execute Outcome	Actions
1. hp core i3	*Label: hp core i3 *Outcome API Name: hp_core_i3	All Conditions Are Met (AND)	Delete Outcome
2. hp core i5	*Label: hp core i5 *Outcome API Name: hp_core_i5		
3. hp core i7			

Below the outcomes, there's a section for the 'Default Outcome'. It includes fields for 'Resource' (\$Record > core type), 'Operator' (Equals), and 'Value' (core i5). There's also a '+ Add Condition' button. Under 'When to Execute Outcome', there are two radio buttons: 'If the condition requirements are met' (selected) and 'Only if the record that triggered the flow to run is updated to meet the condition requirements'. At the bottom right are 'Cancel' and 'Done' buttons.

HP Core i5 Sub-Decision Element:

- Click on the + symbol beside **hp core i5**.
- Select **Decision**.
- **Details Label:** Enter **Months Selected**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **hp 1(i5)**.

- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.how many months.**
- **Operator:** Select **Equals.**
- **Value:** Enter **1.**
- **Click Done.**

Repeat for **hp 2(i5)**, **hp 3(i5)**, **hp 4(i5)**, **hp 5(i5)** with corresponding values **2, 3, 4, 5.**

5. Add Update Records Actions:

For HP Core i5:

- Click on the + symbol after **hp 1(i5)**, **hp 2(i5)**, etc.
- Select **Update Records.**
- **Details Label:** Enter **Update Amount for HP i5.**
- **API Name:** Automatically generated.
- **Field:** Select **Amount__c.**
- **Value:** Set corresponding values:
 - **hp 1(i5):** 1700
 - **hp 2(i5):** 3400
 - **hp 3(i5):** 5100
 - **hp 4(i5):** 6800
 - **hp 5(i5):** 8500
- **Click Done.**

Edit Update Records

one month of hp i5 rate (one_month_of_hp_i5_rate) 

*** How to Find Records to Update and Set Their Values**

Use the laptop bookings record that triggered the flow
 Update records related to the laptop bookings record that triggered the flow
 Use the IDs and all field values from a record or record collection
 Specify conditions to identify records, and set fields individually

Set Filter Conditions

Condition Requirements to Update Record

None—Always Update Record 

Set Field Values for the Laptop Bookings Record

Field	Value	
Amount_c	1700	
+ Add Field		

4) Creating a Flow for Mac Laptop in Salesforce

Objective: To create a Record-Triggered Flow that calculates and updates the amount based on the selected Mac laptop type and the number of months for the laptop booking.

Steps to Create the Flow:

1. Navigate to Flow Builder:

- Go to **Setup**.
- Type "**Flows**" in the Quick Find box.
- Click on **Flows**.
- Click **New Flow**.

2. Select Flow Type:

- Choose **Record-Triggered Flow**.
- Click **Create**.

3. Configure Flow Trigger:

- **Object:** Select **Laptop Booking** from the drop-down list.
- **Trigger Flow When:** Choose **A record is Created or Updated**.
- **Optimize the Flow for:** Select **Actions and Related Records**.
- Click **Done**.

4. Add and Configure Decision Elements:

Decision for Mac Laptop:

- Click on the + symbol next to **mac**.
- Select **Decision**.
- **Details Label:** Enter **Mac Core Type Check**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **Mac Laptop**.
- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.core type**.
- **Operator:** Select **Equals**.
- **Value:** Enter **Bionic Chip**.
- Click **Done**.

Edit Decision

*Label	*API Name												
mac field should be updated	mac_field_should_be_updated												
Description													
Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.													
OUTCOME ORDER 1 mac laptop	OUTCOME DETAILS <table border="1"> <tr> <td>*Label</td> <td>*Outcome API Name</td> </tr> <tr> <td>mac laptop</td> <td>mac_laptop</td> </tr> </table>	*Label	*Outcome API Name	mac laptop	mac_laptop								
*Label	*Outcome API Name												
mac laptop	mac_laptop												
Default Outcome <table border="1"> <tr> <td colspan="3">Condition Requirements to Execute Outcome</td> </tr> <tr> <td colspan="3">All Conditions Are Met (AND)</td> </tr> <tr> <td style="width: 30%;"> Resource <input type="text" value="\$Record > core type X"/> </td> <td style="width: 10%; text-align: center;"> Operator <input type="text" value="Equals"/> </td> <td style="width: 60%;"> Value <input type="text" value="Bionic chip"/> </td> </tr> <tr> <td colspan="3"> <input type="button" value="Add Condition"/> </td> </tr> </table>		Condition Requirements to Execute Outcome			All Conditions Are Met (AND)			Resource <input type="text" value="\$Record > core type X"/>	Operator <input type="text" value="Equals"/>	Value <input type="text" value="Bionic chip"/>	<input type="button" value="Add Condition"/>		
Condition Requirements to Execute Outcome													
All Conditions Are Met (AND)													
Resource <input type="text" value="\$Record > core type X"/>	Operator <input type="text" value="Equals"/>	Value <input type="text" value="Bionic chip"/>											
<input type="button" value="Add Condition"/>													

○

Mac Bionic Chip Sub-Decision Element:

- Click on the + symbol beside **Mac Laptop**.

- **Select Decision.**
- **Details Label:** Enter **Mac Months Selected**.
- **API Name:** Automatically generated.
- **Outcome Details Label:** Enter **Mac Bionic Chip 1 Month**.
- **Outcome API Name:** Automatically generated.
- **Resource:** Select **Record.how many months**.
- **Operator:** Select **Equals**.
- **Value:** Enter **1**.
- **Click Done.**

Repeat for **Mac Bionic Chip 2 Months**, **Mac Bionic Chip 3 Months**, **Mac Bionic Chip 4 Months**, and **Mac Bionic Chip 5 Months** with corresponding values **2, 3, 4, 5**.

OUTCOME ORDER	OUTCOME DETAILS	Delete Outcome
mac bionic chip(1)	*Label mac bionic chip(1)	*Outcome API Name mac_bionic_chip_1
mac bionic chip(2)		
mac bionic chip(3)		
mac bionic chip(4)		
mac bionic chip(5)		
Default Outcome		
+ Add Condition		
When to Execute Outcome <input checked="" type="radio"/> If the condition requirements are met <input type="radio"/> Only if the record that triggered the flow to run is updated to meet the condition requirements		
		Cancel Done

5. Add Update Records Actions:

For Mac Bionic Chip:

- Click on the + symbol after **Mac Bionic Chip 1 Month**, **Mac Bionic Chip 2 Months**, etc.
- Select **Update Records**.
- **Details Label:** Enter **Update Amount for Mac Bionic Chip**.
- **API Name:** Automatically generated.

- **Field:** Select **Amount__c**.
- **Value:** Set corresponding values:
 - **1 Month:** 1700
 - **2 Months:** 3400
 - **3 Months:** 5100
 - **4 Months:** 6800
 - **5 Months:** 8500
- Click **Done**.

Edit Update Records

*** How to Find Records to Update and Set Their Values**

- Use the laptop bookings record that triggered the flow
- Update records related to the laptop bookings record that triggered the flow
- Use the IDs and all field values from a record or record collection
- Specify conditions to identify records, and set fields individually

Set Filter Conditions

Condition Requirements to Update Record

None—Always Update Record ▾

Set Field Values for the Laptop Bookings Record

Field	Value
Amount__c	2000

Cancel Done

6. Save and Activate the Flow:

- Click **Save**.
- **Label:** Enter **Laptop Distributions**.
- **API Name:** Automatically filled.
- Click **Save**.
- Click **Activate** to make the flow live.



○

Topic 11: Creating a Trigger in Salesforce

Objective: To create a trigger that sends an email notification after a laptop booking record is inserted or updated.

Overview: A trigger in Salesforce is used to execute custom code before or after a data manipulation operation (DML event) like insert, update, or delete. Here, we will create an "After Insert" and "After Update" trigger for the Laptop_Bookings__c object to send email notifications to users.

Types of Triggers:

- **Before Triggers:** Used to perform validations or modifications before the record is saved to the database.
- **After Triggers:** Used to perform operations that require the record to be saved to the database first, such as sending email notifications.

Steps to Create a Trigger

1. Open Developer Console:

- Navigate to the gear icon in the top right corner of Salesforce.
- Click on **Developer Console**.

2. Create a New Trigger:

- In Developer Console, click on **File** in the toolbar.
- Select **New** and then **Trigger**.

3. Define Trigger Details:

- **Trigger Name:** LaptopBooking
- **Object:** Laptop_Bookings__c
- **Trigger Event:** Select **After Insert** and **After Update**.

4. Write Trigger Code:

- Paste the following code snippet into the trigger editor:

apex

Copy code

```
trigger LaptopBooking on Laptop_Bookings__c (after insert, after update) {
```

```
    if (Trigger.isAfter && (Trigger.isInsert || Trigger.isUpdate)) {
```

```
LaptopBookingHandler.sendEmailNotification(Trigger.new);  
}  
  
}
```

- **Explanation:**
 - Trigger.isAfter ensures that the trigger runs after the record is inserted or updated.
 - Trigger.isInsert and Trigger.isUpdate check if the operation is an insert or update.
 - LaptopBookingHandler.sendEmailNotification(Trigger.new) calls the handler class method to send an email.

5. Save the Trigger:

- Click **Save** to save the trigger.

Create the Handler Class

1. Create a New Apex Class:

- In Developer Console, click on **File** in the toolbar.
- Select **New** and then **Apex Class**.

2. Define Handler Class Code:

- Paste the following code snippet into the class editor:

apex

Copy code

```
public class LaptopBookingHandler {
```

```
    public static void sendEmailNotification(List<Laptop_Bookings__c> lapList) {  
        for (Laptop_Bookings__c lap : lapList) {  
            Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();  
            email.setToAddresses(new List<String>{lap.Email__c});  
            email.setSubject('Welcome to our company');
```

```

String body = 'Dear ' + lap.Name + ', \n';
body += 'Welcome to Laptop Rentals! You have been seen as a valuable customer to
us.\n';
body += 'Please continue your journey with us, while we try to provide you with good
quality resources. \n';
body += 'Laptop Amount = ' + lap.Amount__c + '\n';
body += 'Core Type = ' + lap.Core__c + '\n';
body += 'Laptop Type = ' + lap.Laptop_Type__c;

email.setPlainTextBody(body);
Messaging.sendEmail(new List<Messaging.SingleEmailMessage>{email});

}
}

}

```

- **Explanation:**

- sendEmailNotification method takes a list of Laptop_Bookings__c records.
- For each record, it creates and sends a personalized email using Messaging.SingleEmailMessage.

3. Save the Handler Class:

- Click **Save** to save the class.

Testing

1. Create Records:

- Add at least 10-12 records to the Laptop_Bookings__c object.
- Ensure to include different types like dell(i3), dell(i7), acer(i3), hp(i5), and mac(bionic chip).

2. Verify Email Notifications:

- Verify that email notifications are sent correctly based on the records inserted or updated.

Documentation

Trigger and Handler Class Details:

- **Trigger Name:** LaptopBooking
- **Trigger Object:** Laptop_Bookings__c
- **Trigger Events:** After Insert, After Update

Handler Class:

- **Class Name:** LaptopBookingHandler
- **Method:** sendEmailNotification
- **Function:** Sends an email notification to users based on the details in the Laptop_Bookings__c records.

Topic 12: Salesforce Reports

Reports in Salesforce provide a way to analyze and visualize your data. They come in various formats, each suited to different types of analysis. Here's an overview of the types of reports you can create in Salesforce:

Types of Reports in Salesforce

1. Tabular Reports

- **Description:** Provides a simple list of data without any subtotals.
- **Use Case:** Ideal for generating straightforward lists or simple data tables.
- **Example:** A report listing all accounts, contacts, or opportunities.

2. Summary Reports

- **Description:** Offers data with groupings and subtotals.
- **Use Case:** Useful for reports that need to show subtotals based on a particular field, or to organize data hierarchically.
- **Example:** A report showing all opportunities grouped and subtotalized by Sales Stage and Owner.

3. Matrix Reports

- **Description:** Allows data to be grouped by both rows and columns, providing a comparison of totals across two dimensions.
- **Use Case:** Best for complex data analysis where you need to view data by multiple dimensions.
- **Example:** A report summarizing opportunities by month (rows) and by account (columns).

4. Joined Reports

- **Description:** Combines multiple report blocks into a single report, each block with unique columns, summaries, formulas, and filters.
- **Use Case:** Suitable for showing related data from different report types or objects in a single view.
- **Example:** A report combining opportunity, case, and activity data for your accounts.

How to Create and Manage Reports

1. Creating a Report:

- Navigate to the **Reports** tab in Salesforce.
- Click **New Report**.
- Choose the report type (Tabular, Summary, Matrix, or Joined).
- Select the object you want to report on and configure the report criteria.
- Add fields, apply filters, and customize the layout as needed.
- Click **Save** and provide a name for your report.

2. **Reading and Analyzing Reports:**

- Use the report view to analyze data, with options to sort, filter, and drill down into details.
- Summary and Matrix reports allow you to view subtotals and comparisons across different dimensions.

3. **Scheduling Reports:**

- Open the report you want to schedule.
- Click **Subscribe** or **Schedule Future Runs**.
- Set the frequency (daily, weekly, monthly) and specify the recipients.
- Save the schedule to automatically run the report and send it to the designated recipients.

4. **Sharing Reports:**

- Reports can be shared by setting permissions and sharing settings.
- You can control who can view or modify the report by adjusting the report folder's sharing settings.

Best Practices

- **Choose the Right Report Type:** Select the type of report that best fits your analysis needs (e.g., use Matrix reports for multi-dimensional data analysis).
- **Utilize Filters:** Apply filters to narrow down the data to only what is relevant for your analysis.
- **Group and Summarize:** Use summary and matrix reports to aggregate data and get insightful summaries.
- **Schedule Regular Reports:** Automate report generation and distribution to ensure timely updates for stakeholders.

By understanding and leveraging these report types, you can effectively analyze and manage your Salesforce data, helping you make more informed decisions and drive business success.

Creating a Report in Salesforce

Here's a step-by-step guide to creating a report in Salesforce, specifically using the report type "Consumer with Laptop Bookings and Total Laptops":

Steps to Create a Report

1. Access the Reports Tab:

- Navigate to the app you're working in.
- Click on the **Reports** tab to open the Reports page.

2. Create a New Report:

- Click on **New Report**.

3. Select the Report Type:

- In the report type selection panel, either:
 - Search for "Consumer with Laptop Bookings and Total Laptops" using the search bar, or
 - Select it from the category or report type panel.
- Click **Start Report** to proceed.

4. Customize Your Report:

- You'll be directed to the report builder interface where you can customize the report.
- Add fields to your report from the left pane. Drag and drop the desired fields into the report layout.

5. Group Rows and Columns:

- Use the grouping options to organize your data:
 - Drag fields into the **Group Rows** or **Group Columns** sections as needed.
 - This will help in summarizing and analyzing the data based on different dimensions.

6. Apply Bucket Fields:

- Click the column drop-down arrow where you want to apply a bucket field.

- Select **Bucket Field** from the options.
- Configure the bucket field:
 - Define the bucket categories (e.g., price ranges, status types).
 - Assign field values to these buckets.
- Click **Apply** to add the bucket field to your report.

7. Save or Run the Report:

- Once you've customized the report and applied any necessary bucket fields, you can:
 - Click **Save** to save the report for future use.
 - Click **Run** to generate and view the report data immediately.

By following these steps, you'll be able to create a customized report that combines data from multiple objects and presents it in a format that meets your needs.

Sharing a Report with the Owner in Salesforce

To share a report with the owner and set up email notifications, follow these steps:

1. Edit the Report:

- Navigate to the report you want to share.
- Click on the **Edit** drop-down menu for the report.

2. Select the Subscribe Option:

- Choose the **Subscribe** option from the drop-down menu.

3. Configure the Subscription:

- In the subscription settings:
 - Set the report to run as the desired frequency (e.g., daily).
 - For the **Run Report As** option, select **Another Person**.

4. Select the Recipient:

- Choose the recipient for the email notifications:
 - Select your personal account or the specific owner who should receive the email notifications.
- This will ensure that the designated owner gets daily email notifications of the report.

5. Save the Subscription:

- Click **Save** to finalize the subscription settings.

Note: The owner will receive daily email notifications containing the laptop booking report, allowing them to view the data remotely.

By setting up this subscription, you ensure that the owner stays updated with the latest report data without having to manually check the report each time.

The screenshot shows a reporting interface with the following details:

Report Title: Report: consumer with Laptop Bookings and Total Laptops
Report Name: Consumer with laptops and total laptops

Total Records: 6 **Total Amount:** ₹63,700

Table Headers: type of versions ↑, consumer: consumer_name ↴, Laptop Bookings: Laptop Bookings ↴, Total No Of Laptops: Total Laptops ↴, Laptop names ↴, core type ↴, Amount ↴

Data Rows:

	consumer: consumer_name	Laptop Bookings: Laptop Bookings	Total No Of Laptops: Total Laptops	Laptop names	core type	Amount
basic (1)	kiran	goodends	hp	Dell	core i3	₹1,200
Subtotal						₹1,200
intermediate (1)	Ramya	camios	acer	Dell	Core i7	₹1,500
Subtotal						₹1,500
high (3)	ravi	manvials	acer	Hp	Core i5	₹4,500
	kiran	workings	hp	Hp	Core i5	₹6,000
	Ramya	laptops	mac	Acer	core i3	₹5,500
Subtotal						₹16,000
very high (1)	vignesh	smartbrids	mac	Hp	Core i7	₹45,000
Subtotal						₹45,000
Total (6)						₹63,700

Bottom Controls: Row Counts, Detail Rows, Subtotals, Grand Total

Topic 13: Salesforce Dashboards

Dashboards in Salesforce are visual representations of your data, created from reports. They allow you to monitor key metrics and trends in a visually appealing way, helping you make informed business decisions based on real-time data.

Basics of Dashboards

1. Components:

- **Charts:** Visualize data with pie charts, bar charts, line charts, etc.
- **Graphs:** Show trends over time or comparisons between different data sets.
- **Gauges:** Display progress toward a goal or target.
- **Tables:** Present data in a grid format for detailed examination.
- **Metrics:** Highlight key performance indicators (KPIs) and summary numbers.

2. Dashboard Filters:

- **Global Filters:** Allow users to apply filters across multiple components within the dashboard.
- **Component Filters:** Apply filters to specific components, enabling more focused analysis.

3. Data Sources:

- Dashboards pull data from reports, which need to be created and configured before building a dashboard.

4. User Access:

- **Sharing:** Dashboards can be shared with users or groups to ensure that relevant stakeholders have access to the information.

5. Refresh Settings:

- Dashboards can be set to refresh at specific intervals to ensure that the data displayed is up-to-date.

Creating a Dashboard Folder

To organize your dashboards effectively, you can create folders to group related dashboards. Here's how to create a new dashboard folder in Salesforce:

1. Navigate to Dashboards:

- Click on the **App Launcher** (grid icon) and type **Dashboards** in the search box.
- Click on the **Dashboards** tab to go to the Dashboards home page.

2. Create a New Folder:

- Click on the **New Folder** button.

3. Enter Folder Details:

- **Folder Label:** Enter Total Rent Amount as the folder label.
- **Folder Unique Name:** This will be auto-populated based on the folder label.

4. Save the Folder:

- Click **Save** to create the new folder.

Now you have a dedicated folder for organizing dashboards related to the total rent amount. You can move or create dashboards within this folder to keep your reports and visualizations well-organized.

Creating a Dashboard

To visualize data effectively, you can create a dashboard in Salesforce and add various components to it. Here's how to create a dashboard and add components:

1. Navigate to Dashboards:

- Go to the **App Launcher** (grid icon) and search for **Dashboards**.
- Click on the **Dashboards** tab to go to the Dashboards home page.

2. Create a New Dashboard:

- Click on the **New Dashboard** button.

3. Enter Dashboard Details:

- **Name:** Enter a name for the dashboard (e.g., Laptop Rental Overview).
- **Folder:** Select the folder you created earlier, such as Total Rent Amount.

4. Create the Dashboard:

- Click **Create** to proceed.

5. Add Components to the Dashboard:

- Click on the **Add Component** button to start adding components.

6. Select a Report:

- Choose a report from the list that you want to use for the component.

- Click **Select** to add the report.

7. Choose Component Type:

- Select the type of component you want to add (e.g., bar chart, pie chart, line graph) based on the report data.

8. Customize the Component:

- Configure the component settings as needed (e.g., title, chart type, data display options).

9. Add to Dashboard:

- Click **Add** to include the component in your dashboard.

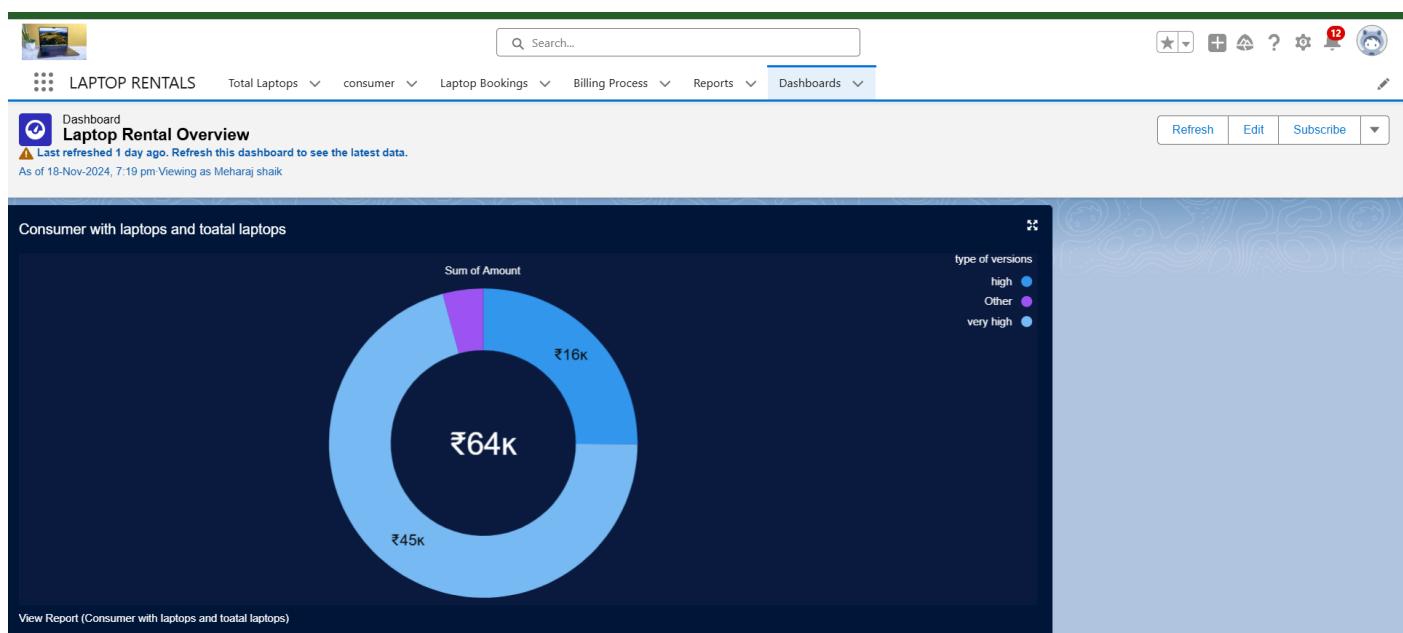
10. Save the Dashboard:

- Click **Save** to save your changes.

11. Finalize:

- Click **Done** to complete the dashboard creation.

Your dashboard will now display the selected report data in the chosen visual format, helping you analyse and interpret your data effectively.



THANK YOU